Individuality transfer: Predicting human decision-making across tasks

Hiroshi Higashi

March 26, 2025

Abstract

Predicting an individual's behaviour in one task based on their behaviour in a different task is a key challenge in modeling individual decision-making tendencies. We propose a novel framework that addresses this challenge by leveraging neural networks and introducing a concept we term the "individuality index." This index, extracted from behaviour in a "source" task via an encoder network, captures an individual's unique decision-making tendencies. A decoder network then utilizes this index to generate the weights of a task-specific neural network (a "task solver"), which predicts the individual's behaviour in a "target" task. Notably, this prediction does not require behavioural data from the target task for that individual. We demonstrate the effectiveness of our approach in two distinct decision-making tasks: a value-guided task and a perceptual task. Our framework provides a robust and generalizable method for parameterizing individuality, offering new insights into individual differences in decision-making.

1 Introduction

Humans (and other animals) exhibit substantial commonalities in their decision-making processes. However, considerable variability is also frequently observed in how individuals perform perceptual and cognitive decision-making tasks [5, 1]. This variability arises from differences in underlying cognitive mechanisms. For example, individuals may vary in their ability to retain past experiences [12, 8], respond to events with both speed and accuracy [51, 44], or explore novel actions [17]. If these factors can be meaningfully disentangled, they would enable a concise characterization of individual decision-making processes, yielding a low-dimensional, parameterized representation of individuality. Such a representation could, in turn, be leveraged to predict future behaviours at an individual level. Shifting from population-level predictions to an individual-based approach would mark a significant advancement in domains where precise behaviour prediction is essential, such as social and cognitive sciences. Beyond prediction, this approach offers a framework for parametrizing and clustering

individuals, thereby facilitating the visualization of behavioural heterogeneity, which has applications in psychiatric analysis [31, 10].

Cognitive modelling is a standard approach for reproducing and predicting human behaviour [29, 3, 56], often implemented within a reinforcement learning framework (e.g., [30, 9, 53]). However, because these cognitive models are manually designed by researchers, their ability to accurately fit behavioural data may be limited [16, 43, 28]. A data-driven approach using artificial neural networks (ANNs) offers an alternative [11, 33, 39]. Unlike cognitive models, which rely on predefined behavioural assumptions [37], ANNs require minimal prior assumptions and can learn complex patterns directly from data. For instance, convolutional neural networks (CNNs) have successfully replicated human choices and reaction times in various visual tasks [25, 35, 15]. Similarly, recurrent neural networks (RNNs) [41, 7] have been applied to model value-guided decision-making tasks such as the multi-armed bandit problem [55, 10]. A promising approach to capturing individual decision-making tendencies while preserving behavioural consistency is to tune ANN weights using a parameterized representation of individuality.

This idea was first proposed by Dezfouli et al. [10], who employed an RNN to solve a two-armed bandit task. Their study utilized an autoencoder framework [38, 47], in which behavioural recordings from a single session of the bandit task, performed by an individual, were fed into an encoder. The encoder produced a low-dimensional vector, interpreted as a latent representation of the individual. Similar to hypernetworks [20, 22], a decoder then took this low-dimensional vector as input and generates the weights of the RNN. This framework successfully reproduced behavioural recordings from other sessions of the same bandit task while preserving individual characteristics. However, since this individuality transfer has only been validated within the bandit task, it remains unclear whether the extracted latent representation captures an individual's intrinsic tendencies across a variety of tasks.

To address this question, we aim to make the low-dimensional representation—referred as *individuality index*—robust to variations across individuals and tasks, thereby enhancing its generalizability. Specifically, we propose a framework that predicts an individual's behaviours, not only in the same task but also in similar yet distinct tasks and environments. If the individuality index indeed serves as a low-dimensional representation of an individual's decision-making process, then extracting it from one task could should facilitate the prediction of that individual's behaviours in another.

In this study, we define the problem of *individuality transfer across tasks* as follows (also illustrated in Figure 1). We assume access to a behavioural dataset from multiple individuals performing two tasks: a *source task* and a *target task*. We train an encoder that takes behavioural data from the source task as input and outputs an individuality index. This index is then fed into a decoder, which generates the weights of an ANN, referred to as a *task solver*, that reproduces behaviours in the target task. For testing, a new individual provides behavioural data from the source task, allowing us to infer their individuality index. Using this index, a task solver is constructed to predict how the test individual will

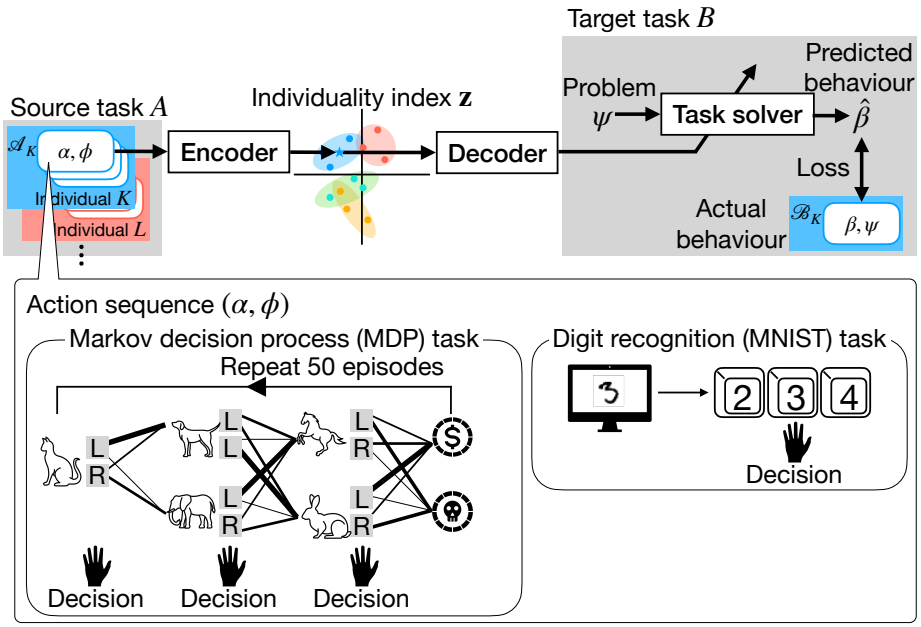

Figure 1: The EIDT (encoder, individuality index, decoder, and task solver) framework for individuality transfer across tasks. The encoder maps action(s) α, provided by an individual K performing a specific problem ϕ in the source task A, into an individuality index (represented as a point in the two-dimensional space in the center). The individual index is then fed into the decoder, which generates the weights for a task solver. The task solver predicts the behaviour of the same individual K in the target task B. During the training, a loss function evaluates the discrepancy between the predicted behaviour $\hat{\beta}$ and the actual recorded behaviour β of individual K. The encoder's input is referred to as an *action sequence*, the form of which depends on task. For example, in a sequential Markov decision process (MDP) task, an action sequence consists of an environment (state transition probabilities) and a sequence of actions over multiple episodes. For a digit recognition task, it consists of a stimulus digit image and the corresponding chosen response.

behave in the target task. Importantly, this prediction does not require any behavioural data from the test individual performing the target task. We refer to this framework as *EIDT*, an acronym for encoder, individuality index, decoder and task solver.

We evaluated whether the proposed EIDT framework can effectively transfer individuality in both value-guided sequential decision-making tasks and perceptual decision-making tasks. To assess its generalizability across individuals, meaning its ability to predict the behaviour of previously unseen individuals, we tested the framework using a test participant pool that was not included in the dataset used for model training. To determine how well our framework captures each individual's unique behavioural patterns, we compared the prediction performance of a task solver specifically designed for a given individual with the performance of task solvers designed for other individuals. Our results indicate that the proposed framework successfully mimics decision-making while accounting for individual differences.

2 Results

We evaluated our individuality transfer framework, which consists of an encoder, a decoder, a task solver, and an individuality index, using two transfer problems: a value-guided sequential decision-making task (an MDP task) and a perceptual decision-making task (a handwritten digit recognition task, referred to as the MNIST task). The dataset for both tasks include two or more distinct experimental settings. Since each human participant completed all settings, we tested transfer problems in which one setting served as the source task, while another served as the target task.

2.1 Markov decision process (MDP) task

The dataset consisted of behavioural data from 98 participants, as detailed in Section 4.2.2. Participants performed the MDP task under two different conditions: a 2-step task and a 3-step task. Each participant completed three sequence blocks for both the 2- and 3-step tasks, with each sequence comprising 50 episodes. Thus, the dataset contained a total of $N = 6$ (sequences) $\times 98$ (participants) $= 588$ (action sequences). For model training, sequence data from 70% of the participants (68 participants) were used to train the encoder and decoder. An additional 10% (10 participants) were designated as validation data, used to determine the optimal number of training iterations. The remaining 20% (20 participants) were allocated for evaluating the performance of the model.

Neural network exhibits superior performance in behaviour prediction. We evaluated our model under two experimental situations, summarized in Table 1. The first situation, Situation SX (no transfer), does not involve individuality transfer across tasks. Since individuality is not explicitly considered

Table 1: Simulated experimental situations. The "Availability" column indicates data availability for a given participant pool and task during model training. In the "Participant pool" column, "train", "valid", and "test" refer to the training, validation, and test participant pools, respectively.

Situation	Participant pool	Availability	
		Source (task A)	Target (task B)
SX	train-valid		Available
	test		To be predicted
SY	train-valid	Available	Available
	test	Available	To be predicted

in this situation, we designed average models as baselines, described below.

Cognitive model (CG) We implemented a cognitive model based on Q-learning, as detailed in Section 4.2.3. The model parameters—including the learning rate q_{lr}, forgetting factor q_{ff}, discount rate q_{dr}, and inverse temperature q_{it}—were estimated using a maximum likelihood approach based on behavioural data from the target task in the training and validation participant pools. Using these estimated parameters, we constructed a Q-learning agent to predict the behaviour of participants in the test participant pool.

Task solver (TS) We employed a task solver TS($\cdot$), as defined in (1). Instead of using the EIDT framework, the weights of the task solver network were directly trained using data from the training participant pool. To determine the optimal stopping point during training, we used early stopping based on the loss measured on the validation participant pool—training was terminated when the validation loss reached its minimum. The trained task solver was then used to predict the behaviour of participants in the test participant pool.

For further details on the model training procedure, refer to Section 4.2.4.

Under Situation SX, we evaluated whether a neural network-based behaviour prediction model (i.e., the task solver) outperforms the cognitive model in predicting average performance across individuals. The result (Figure 2A) confirmed this. A two-way (model: CG/TS, task: 2/3) repeated-measures (RM) ANOVA with Greenhouse-Geisser correction (significant level was 0.05) revealed a significant effect of model ($F_{1,19} = 29.944$, $p < 0.001$, $\eta_p^2 = 0.182$), indicating that the task solver provided significantly better predictions than the cognitive model. We found no significant effect of task ($F_{1,19} = 0.162$, $p = 0.692$, $\eta_p^2 = 0.003$) and no significant interaction effect between model and task ($F_{1,19} = 3.144$, $p = 0.092$, $\eta_p^2 = 0.011$).

EIDT framework excels in individuality transfer. Under Situation SY (transfer across tasks), we assumed that that behavioural data from the source

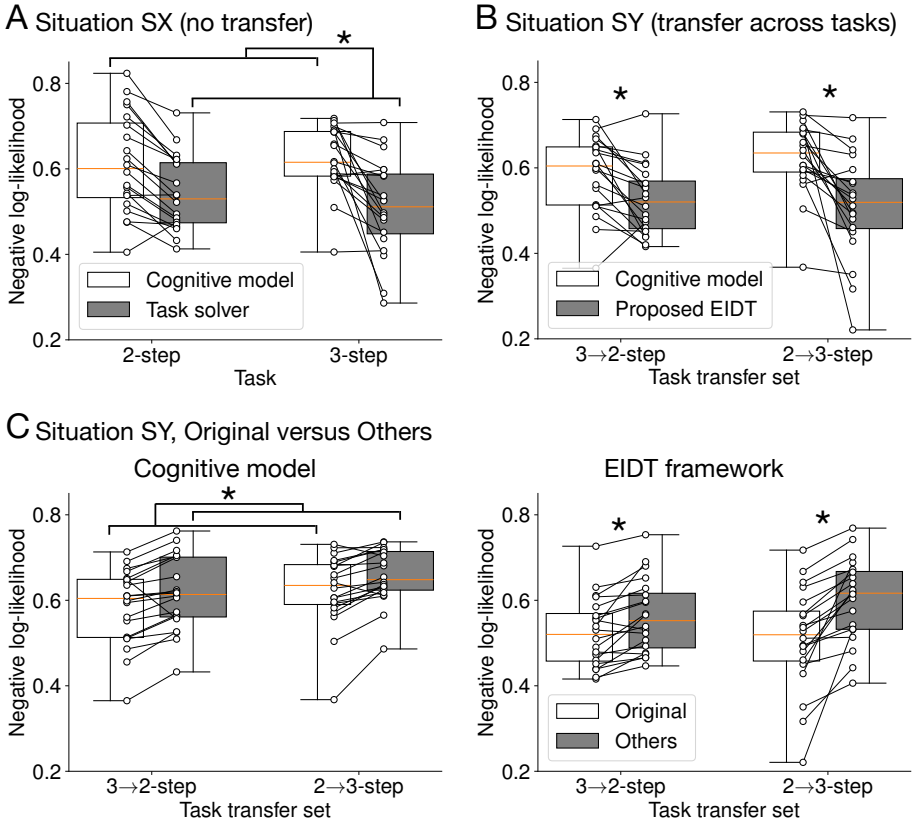

Figure 2: Prediction error results for the MDP task. **A** Under Situation SX. The box represents the interquartile range (IQR), with the central line indicating the median. Whiskers extend to the minimum and maximum values. Connected dots represent data from the same participant. **B** Under Situation SY. **C** Comparison between the transfer-to-original participant (Original) and transfer-to-other-participants (Others) settings.

task of the test participant pool could be leveraged to predict behaviour in the target task. This situation was designed to evaluate individuality transfer across tasks, which is the primary focus of this study. We tested two models:

Cognitive model (CG) We implemented a Q-learning-based cognitive model, as described in Section 4.2.3. The parameters (q_{lr}, q_{ff}, q_{dr}, and q_{it}) for a participant in the test pool were estimated using a maximum likelihood approach based on behavioural data from that participant's source task. Using these estimated parameters, we constructed a Q-learning agent capable of solving the target task (not the source task) and used it to predict the participant's behaviour in the target task.

EIDT We employed our EIDT framework, which was trained using data from both and source and target tasks of the training participant pool. As with the TS model under Situation SX, data from the validation participant pool were used for early stopping during training. To predict behaviour in the target task for a participant K in the test participant pool, we first input each sequence from $\mathcal{A}_K$ (behavioural data from participant K in task A) into the encoder. The individuality index for participant K was computed as the average: $\bar{\boldsymbol{z}}_K = \frac{1}{|\mathcal{A}_K|} \sum_{(\alpha,\phi)\in\mathcal{A}_K} \mathrm{ENC}(\alpha, \phi; \Theta_{\mathrm{ENC}})$. The weights of the task solver for that participant were then generated using the decoder output: $\mathrm{DEC}(\bar{\boldsymbol{z}}^K; \Theta_{\mathrm{DEC}})$. The task solver, incorporating these weights, was used to predict behaviour in response to a given problem ψ in $\mathcal{B}_K$.

For details on the model training procedure, refer to Section 4.2.4.

We compared the prediction error between models, as shown in Figure 2B. A two-way (model: CG/EIDT, transfer task set: 2→3/3→2) RM ANOVA revealed significant effects of model ($F_{1,19} = 21.372$, $p < 0.001$, $\eta_p^2 = 0.182$) and a significant interaction effect ($F_{1,19} = 5.053$, $p = 0.037$, $\eta_p^2 = 0.020$). However, there was no significant effect of transfer task set ($F_{1,19} = 0.330$, $p = 0.573$, $\eta_p^2 = 0.004$). Pairwise comparisons indicated that the cognitive model exhibited significantly greater prediction error than the EIDT model for both transfer task sets: 3→2 ($t_{19} = 3.634$, $p = 0.002$ with Bonferroni correction) and 2→3 ($t_{19} = 4.254$, $p < 0.001$). These results demonstrate that the EIDT model predicts human behaviour more accurately—with a lower negative likelihood—than the cognitive model.

Individuality index forms clusters for each individual. We visualized the individuality indices in Figure 3 and evaluated them using the within-individual and between-individual distances, as defined in Section 4.4.1. For both transfer task sets, the within-individual distance was significantly shorter than the between-individual distance ($t_{19} = -5.515$, $p < 0.001$ for 3→2 and $t_{19} = -2.991$, $p = 0.008$ for 2→3). Since the within-individual distance is shorter than the between-individual distance, the individuality indices from the same individual are positioned closer together in the individuality index space, forming a distinct cluster for each individual.

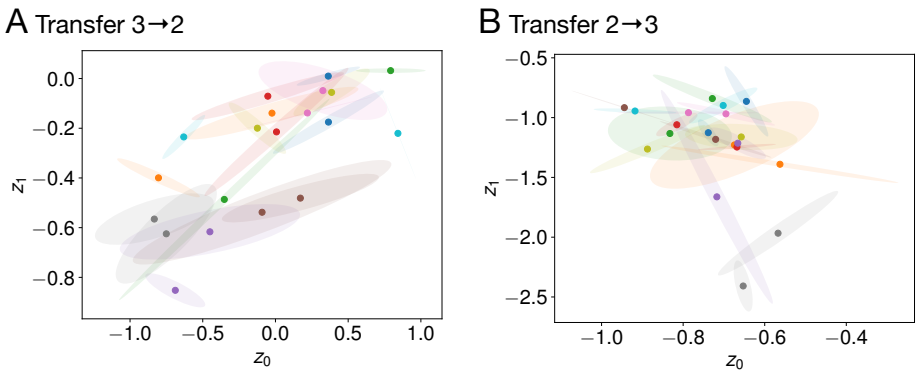

Figure 3: Individuality indices in the test participant pool for the MDP tasks. Dots represent the average individuality index for each participant, while shaded areas show confidence ellipses, enclosing 98.9% of the points for each participant.

We also evaluated the extent to which the variables of the individuality index are disentangled, using the Pearson correlation coefficient. A test of the null hypothesis of uncorrelation showed that the two variables (z_0 and z_1) are significantly correlated for the 3→2 transfer ($r = 0.687$, $p < 0.001$) and the 2→3 transfer ($r = -0.323$, $p = 0.012$).

Individuality is preserved across tasks. We evaluated whether the models could predict unique behavioural patterns for each individual. To do so, we compared prediction accuracy for a given participant's behaviour using a task solver trained specifically for that participant with its accuracy using task solvers trained for other participants. That is, we examined accuracy for the original task solver (Original) versus accuracy for other task solvers (Others). If the task solver effectively captures individuality, it should perform significantly better for Original than for Others. For details on this comparison, refer to Section 4.4.2.

The cognitive model exhibited this tendency (Figure 2C, left). A two-way (transfer task set: 3→2/2→3, transfer-to participant(s): Original/Others) RM ANOVA revealed a significant effect of transfer task set ($F_{1,19} = 5.383$, $p = 0.032$, $\eta_p^2 = 0.060$) and a significant effect of transfer-to participant(s) ($F_{1,19} = 25.169$, $p < 0.001$, $\eta_p^2 = 0.038$), while an interaction effect ($F_{1,19} = 0.047$, $p = 0.831$, $\eta_p^2 < 0.001$) was not significant.

The EIDT model exhibited a similar pattern (Figure 2C, right). A two-way RM ANOVA revealed a significant effect of transfer-to participant(s) ($F_{1,19} = 26.976$, $p < 0.001$, $\eta_p^2 = 0.105$) and an interaction effect ($F_{1,19} = 20.238$, $p < 0.001$, $\eta_p^2 = 0.018$). This did not show a significant effect for transfer task set ($F_{1,19} = 0.201$, $p = 0.659$, $\eta_p^2 = 0.004$). Bonferroni-corrected pairwise comparisons confirmed that prediction loss for Original was significantly lower for Others in both transfer task sets: 3→2 ($t_{19} = 4.278$, $p < 0.001$) and 2→3 ($t_{19} = 5.282$, $p < 0.001$). While the task solver, utilizing the individuality index

of the original participant, predicted that participant's behaviours more accurately than the cognitive model, it was less accurate in predicting the behaviour of other participants. This suggests that the EIDT model successfully captures individual characteristics, leading to personalized behaviour predictions.

2.2 Handwritten digit recognition (MNIST) task

The dataset was originally collected and published by Rafiei et al. [34]. It contains behavioural data from 60 human participants who performed a digit discrimination task. The experiment followed a 2×2 factorial design with two factors: task difficulty (easy versus difficult images, controlled by noise level) and speed pressure (accuracy versus speed focus, controlled by experimental instruction). This design resulted in four experimental settings; EA (easy, accuracy focus), ES (easy, speed focus), DA (difficult, accuracy focus), and DS (difficult, speed focus). Our individuality transfer experiments evaluated whether the proposed framework could transfer individuality from one setting to another. Each setting included 120 unique images, and each participant made a decision for each image twice, resulting in a total of 960 trials per participant. behavioural data from 70% of participants (42 participants) were used to train the encoder and decoder. Additional 10% of participants (6 participants) were assigned as validation data for early stopping during training. The remaining 20% of participants (12 participants) were used to evaluate model performance.

Our average model outperforms a no-fitting model. Situation SX, a no transfer situation, was used to evaluate how well our task solver fits human behaviour compared to an existing model. Ideally, this comparison would involve models that can be explicitly fitted to human behaviour. However, no readily adaptable model was available. As an alternative, we used an average model, RTNet [34], which predicts human-like decisions and reaction times.

RTNet (RN) RTNet accumulates evidence generated by a CNN (see Section 4.3.2) to produce stochastic, human-like decisions and response times.

Task solver (TS) The task solver TS($\cdot$), as defined in Section 4.3, was trained using data from the training participant pool.

Our results confirmed that the task solver with fitting outperforms RTNet without fitting, as shown in Figure 4. A two-way (model: RN/TS, task: EA/ES/DA/DS) RM ANOVA revealed significant effects of model ($F_{3,33} = 104.744$, $p < 0.001$, $\eta_p^2 = 0.694$), task ($F_{1,11} = 322.550$, $p < 0.001$, $\eta_p^2 = 0.876$), and their interaction ($F_{3,33} = 72.733$, $p < 0.001$, $\eta_p^2 = 0.408$). Bonferroni-corrected pairwise comparisons showed significant differences between EA and DA, ES and DA, EA and DS, and ES and DS ($p < 0.001$ for all). However, no significant differences were found between EA and ES or DA and DS. Reproduction accuracy may correlate with the percentage of correct responses in the MNIST task. The original study [34] reported that human accuracy (and RTNets performance) depended on task difficulty but not on focus condition.

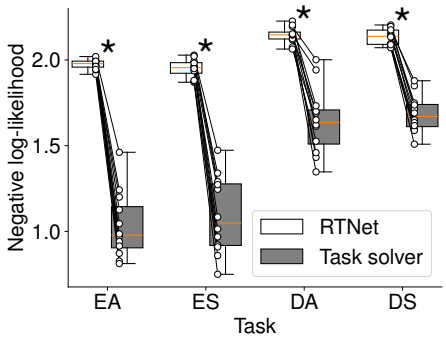

Figure 4: The prediction error under Situation SX (no transfer) in the MNIST task.

In our case, the task solvers achieved correct response rates of 77.7% for EA, 79.0% for ES, 62.0% for DA, and 61.6% for DS. These results are consistent with those reported for RTNet and suggest that reproduction accuracy depends primarily on task difficulty, i.e., uncertainty in response.

Individuality index forms clusters for each individual. We visualized the individuality indices and evaluated them using the within-individual and between-individual distances (see Section S1, Supplementary materials). For most transfer task sets, the within-individual distance was significantly shorter than the between-individual distance, indicating that individuality indices from the same participant tended to cluster together. A test of the null hypothesis of no correlation revealed that the two variables of the individuality index (z_0 and z_1) were significantly correlated for all transfer task sets.

Individuality preserves across tasks. Situation SY (transfer across tasks) was used to evaluate whether our EIDT model could predict the unique behaviour of each individual in an unexperienced (target) task. As in Section 2.1, we compared the prediction accuracy in the target task for a given participant using a task solver trained for that participant (Original) with its accuracy using task solvers trained for other participants (Others).

As shown in Figure 5A, the percentage of correct responses was approximately 78% for the easy difficulty level and 60% for the difficult level, aligning with the accuracy levels of human participants and RTNets [34] A two-way (transfer task set: 12 sets (see x-axis in Figure 5), transfer-to participant(s): Original/Others) RM ANOVA revealed a significant effect of transfer task set ($F_{1,121} = 402.851$, $p < 0.001$, $\eta_p^2 = 0.912$), no significant effect of transfer-to participant(s) ($F_{1,11} = 0.999$, $p = 0.339$, $\eta_p^2 = 0.005$), and no significant interaction effect ($F_{1,121} = 0.478$, $p = 0.718$, $\eta_p^2 = 0.004$).

A comparison using individual behavioural data revealed differences between Original and Others. Figure 5B shows the prediction accuracy for each task

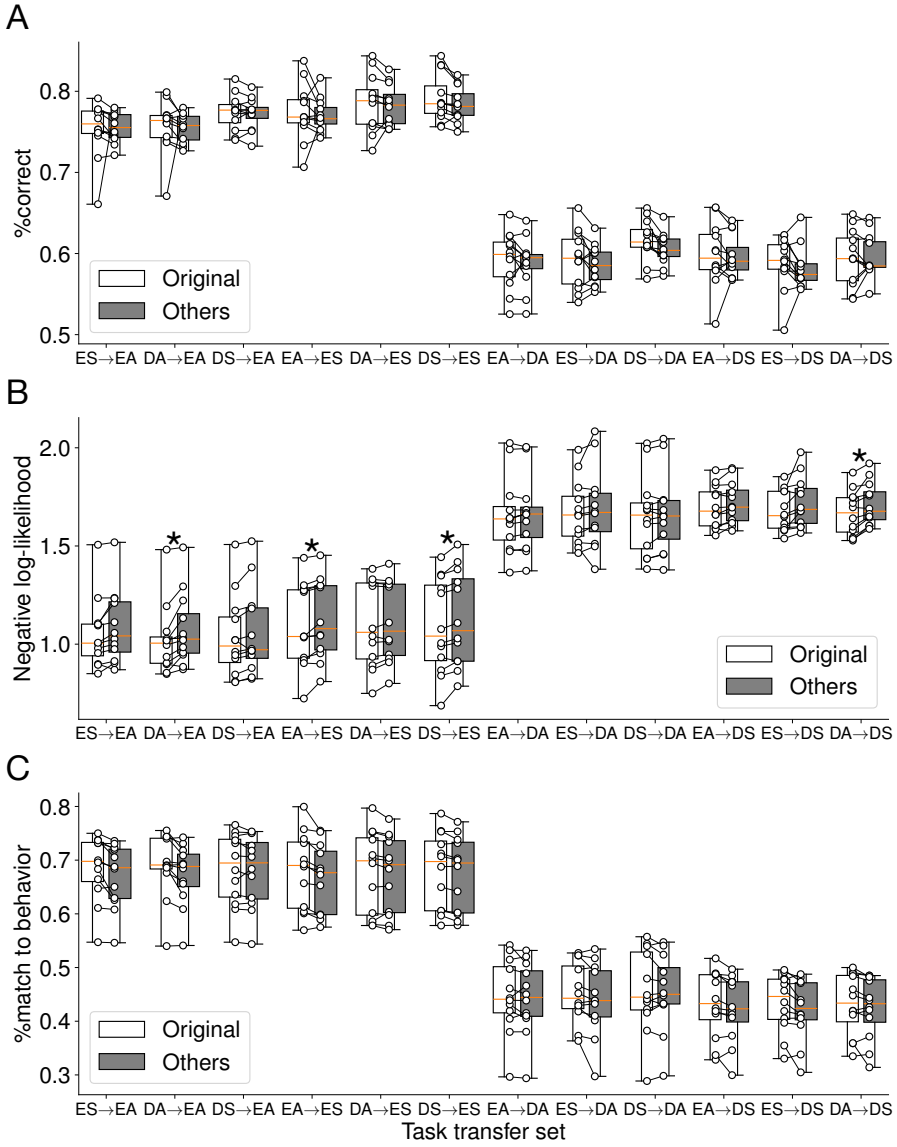

Figure 5: Results for the transfer-to-original-participant (Original) and transfer-to-other-participants (Others) settings in the MNIST task. **A** Percentage of correct responses. **B** Prediction error (likelihood). **C** Percentage of matches to actual behaviour.

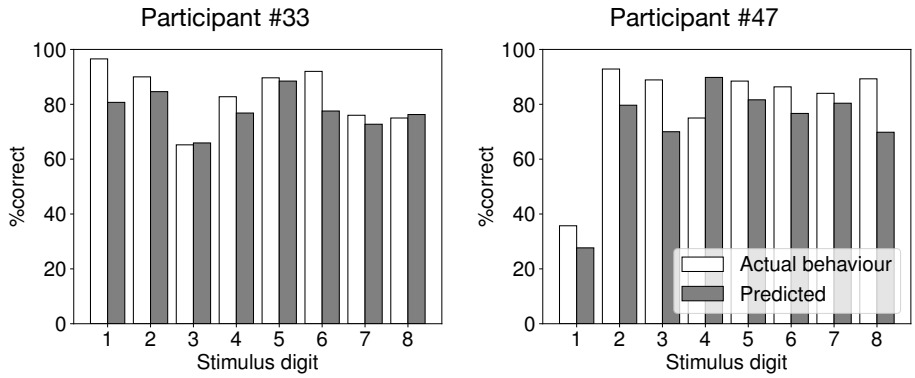

Figure 6: Percentage of correct responses for each stimulus digit in human behaviour and model predictions. Model predictions were conducted under the transfer experiment from Setting ES to Setting EA.

transfer set. A two-way RM ANOVA revealed significant effects of transfer task set ($F_{1,121} = 84.899$, $p < 0.001$, $\eta_p^2 = 0.744$) and transfer-to participant(s) ($F_{1,11} = 18.180$, $p = 0.001$, $\eta_p^2 = 0.006$), and a significant interaction effect ($F_{11,121} = 2.665$, $p = 0.047$, $\eta_p^2 = 0.002$). Bonferroni-corrected pairwise comparisons indicated that the prediction loss for Original was significantly lower than for Others in the transfer task sets of DA→EA, EA→ES, and DS→ES, and DA→DS. However, no significant differences were found in the other eight transfer sets (see Table S1, Supplementary materials). The percentage of matches to actual behaviour is shown in Figure 5C. A two-way RM ANOVA revealed significant effects of transfer task set ($F_{1,121} = 113.049$, $p < 0.001$, $\eta_p^2 = 0.787$) and transfer-to participant(s) ($F_{1,11} = 14.626$, $p = 0.003$, $\eta_p^2 = 0.005$), and no significant interaction ($F_{11,121} = 1.789$, $p = 0.143$, $\eta_p^2 = 0.002$). These results for prediction error and percentage of matches indicate that a task solver specialized for a specific individual using the EIDT model does not predict well the behaviour of other individuals.

Figure 6 visualizes the percentage of correct responses for each digit in the MNIST tasks, highlighting behavioural tendencies reproduced by the model. For example, Participant #33 exhibited a lower percentage for digit 3 compared to other digits, which was captured by the model. Participant #47 showed a drastically lower percentage for digit 1, a pattern also reflected in the model's predictions.

3 Discussion

We proposed an EIDT framework for modeling the unique decision-making process of each individual. This framework enables the transfer of an individuality index from a (source) task to a different (target) task, allowing a task solver predict behaviours in the target task. Several neural network techniques, such as

autoencoders [38, 47], hypernetworks [20], and learning-to-learn [52, 42], facilitate this transfer. Our experiments, conducted on both value-guided sequential and perceptual decision-making tasks, demonstrated the potential of the proposed EIDT framework in individuality transfer across tasks.

EIDT framework transfers individuality across tasks. The EIDT framework was originally proposed by Dezfouli et al. [10] for a bandit task. Their model was designed to predict behaviour within a session performed by a given participant, leveraging behavioural data from other sessions for model training. We extended this idea in three key ways. First, we validated that the framework is effective for previously unseen individuals who were not included in model training. Although these individuals provided behavioural data in the source task to identify their individuality indices, their data were not used for model training. Second, we demonstrated that individuality indices can be transferred even when the source and target tasks differ, indicating that decision-making tendencies are transferable across tasks. Third, while the original work focused on value-guided tasks, we validated the framework's applicability to perceptual decision-making tasks, specifically the MNIST task. These findings establish that the EIDT framework effectively captures individual differences across both tasks and individuals.

Interpreting the individuality index remains challenging. The interpretation of the individuality index remains an open question. Since interpretation often requires task-specific considerations [13], it falls outside the primary scope of this study, whose aim is to develop a general framework for individuality transfer. Previous research [28, 18] has explored associating neural network parameters with cognitive or functional meanings. Approaches such as disentangling techniques [2] and cognitive model integration [19, 48, 43, 14] could aid in better understanding the cognitive and functional significance of the individuality index.

Regarding the individuality index, while disentanglement and separation losses [10] during the training of model training could enhance interpretability, we used only the reproduction loss, as defined in (5). Our results showed that the variables of the individuality index were correlated, suggesting that they do not independently influence behaviour. Although this dependency complicates the interpretation of the individuality index, we decided not to introduce explicit disentanglement constraints because interpretable parameters in cognitive models (e.g., [9]) are not necessarily independent (e.g., an individual with a high learning rate may also have a high inverse temperature [27], resulting these two parameters is represented with one variable).

Why can the encoder extract individuality for unseen individuals? Our experiments, which divided participants into training and test participant pools, demonstrated that the framework successfully extracts individuality for completely new individuals. This generalization likely relies on the fact that in-

dividuality indices form clusters and individuals similar to new participants exist in the training participant pool [56]. The success of individuality transfer in our study suggests that individuals can be clustered based on behavioural patterns. Behavioural clustering has been widely discussed in relation to psychiatric conditions, medication effects, and gender-based differences (e.g., [31, 49, 40]). Our results could contribute to a deeper discussion of behavioural characteristics by clustering not only these groups but also healthy controls.

Which processes contribute to individuality? In the MNIST task, we assumed that individuality emerged primarily from the decision-making process (implemented by an RNN [44, 6]), rather than from the visual processing system (implemented by a CNN [54]). The CNN was pretrained, and the decoder did not tune its weights. Our results do not rule out the possibility that the visual system also exhibits individuality [24, 46]; however, they imply that individual differences in perceptual decision-making can be explained primarily by variations in decision-making system [36, 50, 56, 21]. This assumption provides valuable insights for research on human perception.

Limitations One limitation is that the source and target tasks were relatively similar. Even the cognitive model, despite its lower prediction accuracy, captured individual tendencies to some extent. Thus, our findings do not fully evaluate the generalizability of individuality transfer across diverse task domains. To broaden applicability, future studies should validate transfer across more distinct tasks that share underlying principles of individuality.

The effectiveness of individuality transfer may be influenced by dataset volume. As discussed earlier, prediction performance may depend on whether similar individuals exist in the training participant pool. In our study, 100 participants were sufficient for effective transfer. However, tasks involving greater behavioural diversity may require a substantially larger dataset.

As discussed earlier, the interpretability of the individuality index requires further investigation. Furthermore, the optimal dimensionality of the individuality index remains unclear. This likely depends on the complexity of tasks involved—specifically, the number of factors needed to represent the diversity of behaviour observed in those tasks. While these factors have been explored in cognitive modeling research (e.g., [23]), a clear understanding at the individual level is still lacking. Integrating cognitive modeling with data-driven neural network approaches [10, 19] could help identify key factors underlying individual differences in decision-making.

Future directions To further generalize our framework, a large-scale dataset is necessary, as discussed in the limitations. This dataset should include a large number of participants to ensure prediction performance for diverse individuals [32]. All participants should perform the same set of tasks, which should include a variety of tasks [55]. Building upon our framework, where the encoder

currently accepts action sequences from only a single task, a more generalizable encoder should be able to process behavioural data from multiple tasks to generate a more robust individuality index. To enhance encoder, a multi-head neural network architecture [4] could be utilized. A generalized individuality index would enable transfer to a wider variety of tasks and allow accurate and detailed parameterization of individuals using data from only a single task.

4 Methods

4.1 General framework for individuality transfer across tasks

We formulate the problem of individuality transfer, which involves extracting an individuality index from a source task and predicting behaviour in a target task with preserving individuality. We consider two tasks, A and B, which are different but related. For example, task A might be a 2-step MDP task, while task B is a 3-step MDP task.

The individuality transfer across tasks is defined as follows. An individual K performs a problem within task A, with their behaviour recorded as $\mathcal{A}_K$. Our objective is to predict $\mathcal{B}_K$, which represents K's behaviour when performing task B. To achieve this, we extract an individuality index $\boldsymbol{z}$ from $\mathcal{A}_K$, capturing the individual's behavioural characteristics. This index $\boldsymbol{z}$ is then used to construct a task solver, enabling it to mimic K's behaviour in task B. Since task A provides data for estimating the individuality index and task B is the target of behaviour prediction, we refer to them as the source task and target task, respectively.

Our proposed framework for the individuality transfer consists of three modules:

Task solver predicts behaviour in the target task B.

Encder extracts the individuality index from the source task A.

Decoder generates the weights of the task solver based on the individuality index.

These modules are illustrated in Figure 1. We refer to this framework as EIDT, an acronym for encoder, individuality index, decoder and task solver.

4.1.1 Data representation

For training, we assume that behaviour data from a participant pool $\mathcal{P}$ ($K \notin \mathcal{P}$), where each participant has performed both tasks A and B. These dataset are represented as $\mathcal{A} = \{\mathcal{A}_n\}_{n \in \mathcal{P}}$ and $\mathcal{B} = \{\mathcal{B}_n\}_{n \in \mathcal{P}}$.

For each individual n, the set $\mathcal{A}_n$ consists of one or more sets, each containing a problem instance ϕ (stimuli, task settings, or environment in task A) and a sequence of action(s) α (recorded behavioural responses). For example, in an MDP task, ϕ represents the Markov process (state-action-reward transition)

and α consists of choices over multiple trials. In a simple object recognition task, ϕ is a visual stimulus and α is the participant's response to the stimulus. Similarly, $\mathcal{B}_n$ consists of a problem instance ψ and an action sequence β.

4.1.2 Task solver

The task solver predicts the action sequence for task B as

$$\hat{\beta} = \mathrm{TS}(\psi; \Theta_{\mathrm{TS}}), \tag{1}$$

where ψ is a specific problem in task B and Θ_{TS} represents the solver's weights. The task solver architecture is tailored to task B. For example, in an MDP task, the task solver outputs a sequence of actions in response to ψ. In a simple object recognition task, it produces an action based on a visual stimulus ψ.

4.1.3 Encoder

The encoder processes an action sequence(s) α and generates an individuality index $\boldsymbol{z} \in \mathbb{R}^M$ as

$$\boldsymbol{z} = \mathrm{ENC}(\alpha, \phi; \Theta_{\mathrm{ENC}}), \tag{2}$$

where ϕ is a problem in task A, Θ_{ENC} represents the encoder's weights, and M is the dimensionality of the individual index. The encoder architecture is task-specific and designed for task A.

4.1.4 Decoder

The decoder receives the individuality index $\boldsymbol{z}$ and generates the task solver's weights as

$$\Theta_{\mathrm{TS}} = \mathrm{DEC}(\boldsymbol{z}; \Theta_{\mathrm{DEC}}), \tag{3}$$

where Θ_{DEC} represents the decoder's weights. Since the decoder determines the task solver's weights, it functions as a hypernetwork [20, 22].

4.1.5 Training objective

Although tasks A and B differ, an individual's decision-making system remains consistent across tasks. We model this using the individuality index $\boldsymbol{z}$, linking it to the task solver via the encoder and decoder. For training, we use behavioural dataset $\{\mathcal{A}_n, \mathcal{B}_n\}_{n \in \mathcal{P}}$ from a individual pool $\mathcal{P}$.

Let α be an action sequence representing individual n's behaviour on the source task, i.e., $(\alpha, \phi) \in \mathcal{A}_n$, $n \in \mathcal{P}$. The individuality index is derived by $\boldsymbol{z} = \mathrm{ENC}(\alpha, \phi; \Theta_{\mathrm{ENC}})$. The weights of the task solver are then given by $\Theta_{\mathrm{TS}} = \mathrm{DEC}(\boldsymbol{z}; \Theta_{\mathrm{DEC}})$. Subsequently, the task solver, with the given weights, predicts an action sequence for task B as $\hat{\beta} = \mathrm{TS}(\psi; \Theta_{\mathrm{TS}})$, where $(\beta, \psi) \in \mathcal{B}_n$. We then measure the prediction error between $\hat{\beta}$ and β as:

$$L_p(\alpha, \phi, \beta, \psi, \Theta_{\mathrm{ENC}}, \Theta_{\mathrm{DEC}}) = O(\beta, \hat{\beta}), \tag{4}$$

where β is an action sequence in $\mathcal{B}_n$ recorded along with the problem ψ, and $O(\cdot,\cdot)$ is a suitable loss function (e.g., likelihood-based loss for probabilistic outputs). Using the datasets containing the behaviour of the individual pool $\mathcal{P}$, the weights of the encoder and decoders, Θ_{ENC} and Θ_{DEC}, are optimized by minimizing the total loss:

$$L(\Theta_{\mathrm{ENC}}, \Theta_{\mathrm{DEC}}) = \frac{1}{|\mathcal{P}|} \sum_{n \in \mathcal{P}} \frac{1}{|\mathcal{A}_n|} \sum_{(\alpha,\phi) \in \mathcal{A}_n} \frac{1}{|\mathcal{B}_n|} \sum_{(\beta,\psi) \in \mathcal{B}_n} L_p(\alpha, \phi, \beta, \psi, \Theta_{\mathrm{ENC}}, \Theta_{\mathrm{DEC}}). \tag{5}$$

This section provides a general formulation of individuality transfer across two tasks. For specific details on task architectures and loss functions, see Sections 4.2 and 4.3.

4.2 Experiment on MDP task

We validated our individuality transfer framework using two different decision-making tasks: the MDP task and the MNIST task. This section focuses on the MDP tasks, a dynamic multi-step decision-making task.

4.2.1 Task

At the beginning of each episode, an initial state-cue is presented to the participant. For human participants, the state-cue is represented by animal images (Figure 7). For the cognitive model (Q-learning agent) and neural network-based model, the state-cue is represented numerically (e.g., $(2, 1)$ for the first task state in the second choice). The participant makes a binary decision (denoted as action C_1 or C_2) for each step. In the human experiment, these actions correspond to pressing the left or right cursor key. With a certain probability (either 0.8/0.2 or 0.6/0.4), known as the state-action transition probability, the participant transitions to one of two subsequent task states. This process repeats two times for the 2-step MDP task and three times in the 3-step MDP task. After the final step, the participant receives an outcome: either a reward ($r = 1$) or no reward ($r = 0$). For human participants, rewards were displayed as symbols, as shown in Figure 7. Each sequence from initial state-cue presentation to reward delivery constitutes an *episode*.

The state-action transition probability $T(s, a, s')$ from a task state s to a preceding state s' given an action a varies gradually across episodes. With probability p_{trans}, one of the transition probabilities switches to a new set chosen from $\{(0.8, 0.2), (0.2, 0.8), (0.6, 0.4), (0.4, 0.6)\}$. Consequently, participants must adjust their decision-making strategy in response to these shifts in transition probabilities to maintain reward maximization.

4.2.2 Behavioural data collection

We recruited 123 participants via Prolific. All participants provided their informed consent online. This study was approved by the Committee for Human

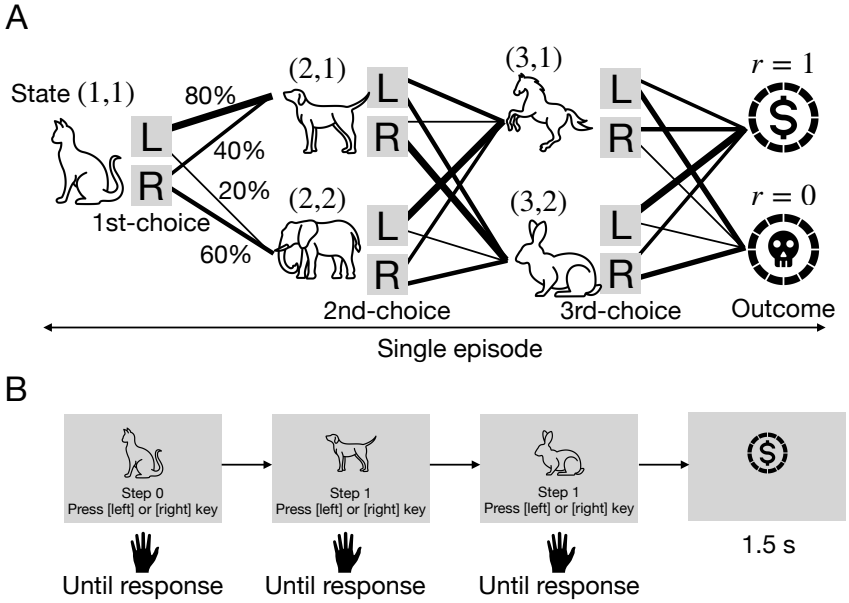

Figure 7: The 3-step MDP task. **A.** Tree diagram illustrating state-action transitions. **B.** Flow of a single episode in the behavioural experiment for human participants.

Research at the Graduate School of Engineering, The University of Osaka, and compiled with the Declaration of Helsinki. Participants received a base compensation of £4 for completing the entire experiment, A performance-based bonus (£0 to £2, average: £1) was awarded based on rewards earned in the MDP task.

Each participant completed 3 sequences for each step condition (2-step and 3-step MDP tasks), with each sequence comprising 50 episodes. The order of the 2-step and 3-step MDP tasks was randomized across sequences. State-cue assignment (animal images) were randomly determined for each sequence. Participants took a mandatory break ($\geq$1 minute) between sequences.

To ensure data quality, we excluded participants based on response time, action bias, and task performance, as follows. Participants with an average response time was below 0.75 s or whose standard deviation of response time was over 10 s were excluded. Additionally, participants exhibiting a strong action bias (where 85% or more of actions were either left or right presses) were rejected. Furthermore, participants with an average reward below below (-1.5 $\times$ interquartile range + first quantile) were excluded. This screening resulted in a final sample of 98 participants.

4.2.3 Cognitive model: Q-learning

To model decision-making in the MDP task, we employed a Q-learning agent [45]. At each step t, the agent was presented with the current task state s_t and selected an action a_t. The agent maintained Q-values, denoted as $Q(s, a)$, for all state-action pairs, where s was a state of the set of all possible task states $\mathcal{S}$ and a was an action of the set of available actions in that state $\mathcal{C}_s$. The probability of selecting action a was determined by a softmax policy:

$$\pi(a) = \frac{\exp^{q_{\mathrm{it}} Q(s_t, a)}}{\sum_{a' \in \mathcal{C}_{s_t}} \exp^{q_{\mathrm{it}} Q(s_t, a')}}, \tag{6}$$

where $q_{\mathrm{it}} > 0$ was a parameter called the inverse temperature or reward sensitivity, controlling the balance between exploration and exploitation.

After selecting action a_t, the agent received an outcome $r_t \in 0, 1$ and transitioned to a new state s_{t+1}. The Q-value for the selected action was updated by

$$Q(s_t, a_t) \leftarrow (1 - q_{\mathrm{lr}}) Q(s_t, a_t) + q_{\mathrm{lr}} (r_t + q_{\mathrm{dr}} \max_{a \in \mathcal{C}_{s_{t+1}}} Q(s_{t+1}, a)), \tag{7}$$

where $q_{\mathrm{lr}} \in (0, 1)$ was the learning rate, determining how much newly acquired information replaced existing knowledge, and $q_{\mathrm{dr}} \in (0, 1)$ was the discount rate, governing the extent to which future rewards influenced current decision. For actions not selected, the agent applied a forgetting factor q_{ff}, updating Q-values as

$$Q(s_t, a) \leftarrow q_{\mathrm{ff}} Q(s_t, a), \tag{8}$$

for $a \in (\mathcal{C}_{s_t} \setminus \{a_t\})$.

4.2.4 EIDT model

This section describes the specific models used for individuality transfer in the MDP task.

Data representation Since MDP tasks involve sequential decision-making, each action sequence consists of multiple actions within a single session. In our experiment, each participant completed L trials per session, with $L = 100$ for the 2-step task and $L = 150$ for the 3-step MDP task. The action sequence is represented as $[(s_1, a_1, r_1), \ldots, (s_L, a_L, r_L)]$, where, s_t denotes the task state at trial t, $a_t \in C$ represents the action selected from the set $C \equiv \{C_k\}_{k=1}^{K}$ (with $K = 2$ in out task), and $r_t \in \{0, 1\}$ indicates whether a reward was received. In the M-step MDP task described in Section 4.2, each task state is represented as (m, c_m), where m denotes the current step within the episode ($m \in \{1, \ldots, M\}$) and c_m corresponds to the cue presented to the participant. The action sequence, denoted as α or β, consists of a sequence of selected actions $(a_1, \ldots, a_L)$, while a problem, denoted as ϕ or ψ, is represented as $((s_1, \ldots, s_L), (r_1, \ldots, r_L))$.

Task solver Before describing the encoder and decoder, we define the architecture of the task solver, which generates actions for the M-step MDP task. The task solver is implemented using a gated recurrent unit (GRU) [7] with Q cells, where $Q = 4$ for the 2-step task and $Q = 8$ for the 3-step task. At time-step t, the GRU takes as input the previous hidden state $\boldsymbol{h}_{t-1} \in \mathbb{R}^Q$, the previous task state s_{t-1}, the previous action a_{t-1}, the previous reward r_{t-1}, and the current task state s_t. It then updates the hidden state as

$$\boldsymbol{h}_t = \mathrm{GRU}(s_{t-1}, a_{t-1}, r_{t-1}, s_t, \boldsymbol{h}_{t-1}; \Phi), \tag{9}$$

where Φ represents the GRU's weights. The updated hidden state is then used to predict the probability of selecting each action through a fully-connected feed forward layer:

$$\boldsymbol{v}_t = \boldsymbol{W}\boldsymbol{h}_t, \tag{10}$$

where $\boldsymbol{v}_t$ represents the logit scores for each action (unnormalised probabilities), and $\boldsymbol{W} \in \mathbb{R}^{K \times Q}$ is the weight matrix. The probabilities of each action are computed using a softmax layer:

$$\pi(a_t = C_k) = \frac{e^{[\boldsymbol{v}_t]_k}}{\sum_{k'=1,\ldots,K} e^{[\boldsymbol{v}_t]_{k'}}}, \tag{11}$$

where $\pi(a_t = C_k)$ represents the probability of selecting action C_k at time t, and $[\boldsymbol{v}_t]_i$ denotes the ith element of $\boldsymbol{v}_t$.

For input encoding, we used a 1-of-K scheme. The step of the MDP task is encoded as $[1, 0, 0]$ for step 1, $[0, 1, 0]$ for step 2, and $[0, 0, 1]$ for step 3. Each task state s_m is represented as $[1, 0]$ or $[0, 1]$ to distinguish the two state cues at each step. The participant's action is encoded as C_1: $[1, 0]$ or C_2: $[0, 1]$, while the reward is represented as 0: $[1, 0]$ or 1: $[0, 1]$. These encodings are concatenated to form input sequences.

The task solver $\mathrm{TS}(\psi; \Theta_{\mathrm{TS}})$ generates a sequence of predicted action probabilities $\{(\pi(a_t = C_1), \ldots, \pi(a_t = C_K))\}_{t=1}^{L}$, using the GRU, the fully-connected layer $\boldsymbol{W}$, and the softmax layer. The problem ψ defines the MDP environment, specifying state transitions and reward outcomes in response to selected action.

To evaluate prediction accuracy, the loss function $O(\beta, \hat{\beta})$, defined in (4), compares human-performed action $\{\beta, \psi\}$ with those predicted by the task solver, $\{\hat{\beta}, \psi\}$. Notably, the problem ψ is not executed with the task solver; instead, the task solver predicts action probabilities based on the same task state and reward history as in the human behavioural data.

Encoder and decoder The encoder $\mathrm{ENC}(\alpha, \phi; \Theta_{\mathrm{ENC}})$ extracts an individuality index $\boldsymbol{z}$ from a sequence of actions α corresponding to a given environment ϕ. The first module of the encoder is a GRU, similar to the task solver, with $R = 32$ cells. The final hidden state $\boldsymbol{h}_L \in \mathbb{R}^R$ serves as the basis for computing the individuality index $\boldsymbol{z} \in \mathbb{R}^M$ using a fully-connected feed-forward network with four layers $d(\cdot)$ as $\boldsymbol{z} = d(\boldsymbol{h}_L)$.

The decoder takes the individuality index $\boldsymbol{z}$ as input and generates the weights for the task solver by $\Theta_{\mathrm{TS}} = \mathrm{DEC}(\boldsymbol{z}; \Theta_{\mathrm{DEC}})$. The decoder is implemented as a single-layer linear network.

4.3 Experiment on MNIST task

This section describes the specific models used for individuality transfer in handwritten digit recognition (MNIST) task.

4.3.1 Task

The dataset used in this experiment was originally collected and published by Rafiei et al. [34]. In this task, participants were presented with a stimulus image depicting a handwritten digit and were required to respond by pressing the corresponding number key, as illusrtaed Figure 1. For further details regarding the task design and data collection, refer to [34].

4.3.2 EIDT model

Data representation An action sequence, denoted as α or β, consists of a single action a and its corresponding response time b. The associated problem, represented as ϕ or ψ, corresponds to a stimulus image. The action a is selected from a set $\{C_1, \dots C_K\}$. Since the task involves recognizing digits ranging from 0 to 9, the number of possible actions is $K = 10$. The stimulus image, ϕ or ψ, is an image of size $H \times W$. In this experiment, we adopted the same resolution as [34], setting $H = W = 227$.

Task solver The task solver for the handwritten digit recognition task is based on the model proposed by [34]. Their model consists of a CNN and an evidence accumulation module. However, since their model represents average human behaviour and does not account for individuality differences, we replace the accumulation module with an RNN [6] to capture individuality. The CNN module processes the input image and produces an evidence vector $\boldsymbol{e} = \mathrm{CNN}(\psi)$, where $\boldsymbol{e} \in \mathbb{R}^K$ and $\mathrm{CNN}(\cdot)$ is based on the AlexNet architecture [26]. The weights of the CNN are sampled from a Bayesian neural network (BNN), introducing istochasticity in the output. This stochasticity enables the models to generate human-like, probabilistic decisions.

The stimulus image is fed into the CNN S times, generating S evidence distributions $\boldsymbol{e}_t \in \mathbb{R}^K$ at each time step $t = 0, \dots, S-1$. In this study, we set $S = 16$ to match the maximum response time, as described later. Since the CNN weights are stochastically sampled from the BNN, the CNN's output varies even when the same image is input multiple times. To model individuality in decision-making, we introduce an RNN with Q cells ($Q = 4$ in our setup). The RNN receives as input the previous hidden state $\boldsymbol{h}_{t-1} \in \mathbb{R}^Q$ and the current evidence $\boldsymbol{e}_t$, updating its hidden state as

$$\boldsymbol{h}_t = f(\boldsymbol{e}_t, \boldsymbol{h}_{t-1}; \Phi), \tag{12}$$

where Φ represents the RNN's network weights. The updated hidden state is passed through a dense layer (as defined in (10)) and a softmax layer (as defined in (11)) to generate the probability distribution over possible digit classifications $[P_t(C_1), \ldots, P_t(C_K)]$ at each time step t.

To evaluate the prediction error, we compare the action sequences generated by human participants $\{\beta, \psi\}$ with those predicted by the task solver $\{\hat{\beta}, \psi\}$, incorporating response times into these analysis. The actual response time b is converted into an integer time step $\tilde{t}$ using the formula: $\tilde{t} = \text{round}(10b)$. For example, a response time of $b = 0.765$ sec is converted to $\tilde{t} = 8$. The likelihood of observed decision is then calculated as $P_{\tilde{t}}(a)$, where a is the actual digit chosen by the participant.

In this task solver, the CNN (driven by BNN) models a visual processing system, while the RNN represents the decision-making system. We assume that the visual system (implemented by CNN and BNN) is shared across all individuals, whereas the decision-making system (implemented by RNN) captures individual differences. Based on this assumption, the CNN and BNN are pretrained using the MNIST dataset [26], and their weight distributions are fixed across individuals. The pretraining procedure followed the original methodology [34].

Encoder and decoder Since each action sequence contains only a single action, it does not form a true "sequence." This makes it challenging to extract individuality from a single data point. To address this, the encoder takes a set of single action sequences as input rather than a single sequence. Specifically, the encoder $\text{ENC}(\{\alpha_u, \phi_u\}_{u=1}^{U}; \Theta_{\text{ENC}})$ extracts the individuality index $\boldsymbol{z}$ from U sets of stimulus images ϕ_u and their corresponding responses α_u, where $u = 1, \ldots, U$. Here, ϕ_u represent the stimulus presented in the uth trial, and α_u represents the corresponding response. The number of action sequences U corresponds to the number of samples available for each individual in the dataset. Since the outputs for these action sequences are just averaged, U can be adjusted flexibly.

The encoder architecture consists of a single CNN module, a single GRU, and a fully-connected feed-forward network. The CNN module is identical to the one used in the task solver. Given an input ϕ_u, let $\boldsymbol{e}_{t,u}$ represent the evidence output from the CNN at time step t. The GRU, which consists R cells ($R = 16$ in our setup), updates its hidden state based on the previous state, the current CNN evidence, and an encoding of the response action by

$$\boldsymbol{h}_{t,u} = \text{GRU}(\boldsymbol{e}_{t,u}, k(a, t, \tilde{t}), \boldsymbol{h}_{t-1,u}; \Psi), \tag{13}$$

where Ψ represents the network weights. The function $k(a, t, \tilde{t})$ outputs the one-hot encoded action a if $t = \tilde{t}$, and zeros otherwise. The value $\tilde{t}$ represents the converted response time, obtained from the original response time b in the action sequence α_u. After processing all U sequences, the final hidden states are averaged across sequences: $\boldsymbol{h} = \frac{1}{U}\sum_{u=1}^{U} \boldsymbol{h}_{L,u}$. The individuality index is then computed as $\boldsymbol{z} = d(\boldsymbol{h})$, where $d(\cdot)$ represents a single-layer fully-connected feed-forward network. The decoder, implemented as a single linear layer, takes the individuality index $\boldsymbol{z}$ as input and outputs the weights for the task solver.

4.4 Analysis

4.4.1 Within- and between-individual distances for evaluating the individuality index

To evaluate the disentanglement of the individuality index, we computed two types of distances: within-individual distance and between-individual-distance. Let $\boldsymbol{z}(\alpha, \phi)$ denote the individuality index derived from an action sequence (α, ϕ). The within-individual distance quantifies the average distance between the individuality index for each action sequence and the average individuality index for a given individual. It is defined as

$$\kappa_n = \frac{1}{|\mathcal{A}_n|} \sum_{(\alpha,\phi)\in\mathcal{A}_n} \|\boldsymbol{z}(\alpha, \phi) - \bar{\boldsymbol{z}}_n\|_2, \tag{14}$$

where $\bar{\boldsymbol{z}}_n$ represents the average individuality index for individual n, computed as $\bar{\boldsymbol{z}}_n = \frac{1}{|\mathcal{A}_n|} \sum_{(\alpha,\phi)\in\mathcal{A}_n} \boldsymbol{z}(\alpha, \phi)$. The between-individual distance measures the distance between the average individuality index for each individual and the overall average individuality index across all individuals. It is defined as

$$\lambda_n = \|\bar{\boldsymbol{z}}_n - \bar{\boldsymbol{z}}\|_2, \tag{15}$$

where $\bar{\boldsymbol{z}}$ represents the average individuality index across all individuals, given by $\bar{\boldsymbol{z}} = \frac{1}{|\mathcal{P}|} \sum_{n\in\mathcal{P}} \bar{\boldsymbol{z}}_n$. If the within-individual distance κ_n is smaller than the between-individual distance λ_n, this suggests that the individuality indices derived from different behavioural data of individual n are closely clustered, indicating consistency in the captured decision-making tendencies. Comparing these distances provides insight into whether the individuality index effectively represents stable and distinctive characteristics of an individual's decision-making process.

4.4.2 Evaluation for uniqueness in task solvers

To assess whether a task solver generates unique behaviour for a specific participant, we compared two prediction loss scores. The first score, referred to as Original, represents the standard prediction loss for a specific participant when using the task solver designed for that participant. It is formulated as

$$L_K = \frac{1}{|\mathcal{A}_K|} \sum_{(\alpha,\phi)\in\mathcal{A}_K} \frac{1}{|\mathcal{B}_K|} \sum_{(\beta,\psi)\in\mathcal{B}_K} L_p(\alpha, \phi, \beta, \psi), \tag{16}$$

where Θ_{ENC} and Θ_{DEC} are omitted for simplicity, as in (4). The second score, referred to as Others, measures the prediction loss for a specific participant when using task solvers designed for other participants. It is formulated as

$$\Upsilon_K = \frac{1}{|\mathcal{P} \setminus \{K\}|} \sum_{n\in(\mathcal{P}\setminus\{K\})} \frac{1}{|\mathcal{A}_n|} \sum_{(\alpha,\phi)\in\mathcal{A}_n} \frac{1}{|\mathcal{B}_K|} \sum_{(\beta,\psi)\in\mathcal{B}_K} L_p(\alpha, \phi, \beta, \psi), \tag{17}$$

where $\mathcal{P}$ represents the set of participant indices in the test participant pool. If the task solvers successfully captures the unique decision-making tendency of each participant, the Original score L_K should be lower than the Others score Υ_K for all participants $K \in \mathcal{P}$.

5 Data availability

The behavioural data for the MDP task has been made publicly available at `https://github.com/hgshrs/indiv_trans`

6 Code availability

All code and trained models have been made publicly available at `https://github.com/hgshrs/indiv_trans`

References

[1] N. J. Boogert, J. R. Madden, J. Morand-Ferron, and A. Thornton. Measuring and understanding individual differences in cognition. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 373(1756):20170280, 9 2018.

[2] C. P. Burgess, I. Higgins, A. Pal, L. Matthey, N. Watters, G. Desjardins, and A. Lerchner. Understanding disentangling in beta-VAE. *arXiv preprint*, (arXiv:1804.03599), 4 2018.

[3] J. R. Busemeyer and J. C. Stout. A contribution of cognitive decision models to clinical assessment: Decomposing performance on the Bechara gambling task. *Psychological Assessment*, 14(3):253–262, 2002.

[4] M. Canizo, I. Triguero, A. Conde, and E. Onieva. Multi-head CNN–RNN for multi-time series anomaly detection: An industrial case study. *Neurocomputing*, 363:246–260, 10 2019.

[5] J. B. Carroll and S. E. Maxwell. Individual differences in cognitive abilities. *Annual Review of Psychology*, 30(1):603–640, 1 1979.

[6] Y.-A. Cheng, I. Felipe Rodriguez, S. Chen, K. Kar, T. Watanabe, and T. Serre. RTify: Aligning deep neural networks with human behavioral decisions. In *Advances in Neural Information Processing Systems*, 2024.

[7] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Stroudsburg, PA, USA, 2014. Association for Computational Linguistics.

[8] A. G. E. Collins and M. J. Frank. How much of reinforcement learning is working memory, not reinforcement learning? A behavioral, computational, and neurogenetic analysis. *European Journal of Neuroscience*, 35(7):1024–1035, 4 2012.

[9] N. D. Daw, S. J. Gershman, B. Seymour, P. Dayan, and R. J. Dolan. Model-based influences on humans' choices and striatal prediction errors. *Neuron*, 69(6):1204–1215, 3 2011.

[10] A. Dezfouli, H. Ashtiani, O. Ghattas, R. Nock, P. Dayan, and S. O. Cheng. Disentangled behavioral representations. In *Advances in Neural Information Processing Systems*, 2019.

[11] A. Dezfouli, K. Griffiths, F. Ramos, P. Dayan, and B. W. Balleine. Models that learn how humans learn: The case of decision-making and its disorders. *PLOS Computational Biology*, 15(6):e1006903, 6 2019.

[12] K. D. Duncan and D. Shohamy. Memory states influence value-based decisions. *Journal of Experimental Psychology: General*, 145(11):1420–1426, 11 2016.

[13] M. K. Eckstein, S. L. Master, L. Xia, R. E. Dahl, L. Wilbrecht, and A. G. Collins. The interpretation of computational model parameters depends on the context. *eLife*, 11:75474, 11 2022.

[14] M. K. Eckstein, C. Summerfield, N. D. Daw, and K. J. Miller. Predictive and interpretable: combining artificial neural networks and classic cognitive models to understand human learning and decision making. In *Proceedings of the 45th Annual Conference of the Cognitive Science Society*, pages 928–935, 2023.

[15] T. Fel, I. Felipe, D. Linsley, and T. Serre. Harmonizing the object recognition strategies of deep neural networks with humans. In *Advances in Neural Information Processing Systems*, 2022.

[16] M. Fintz, M. Osadchy, and U. Hertz. Using deep learning to predict human decisions and using cognitive models to explain deep learning models. *Scientific Reports*, 12(1):4736, 3 2022.

[17] M. J. Frank, B. B. Doll, J. Oas-Terpstra, and F. Moreno. Prefrontal and striatal dopaminergic genes predict individual differences in exploration and exploitation. *Nature Neuroscience*, 12(8):1062–1068, 8 2009.

[18] Y. Ger, E. Nachmani, L. Wolf, and N. Shahar. Harnessing the flexibility of neural networks to predict dynamic theoretical parameters underlying human choice behavior. *PLOS Computational Biology*, 20(1):e1011678, 1 2024.

[19] Y. Ger, M. Shahar, and N. Shahar. Using recurrent neural network to estimate irreducible stochasticity in human choice behavior. *eLife*, 13, 9 2024.

[20] D. Ha, A. Dai, and Q. V. Le. Hypernetworks. *arXiv preprint*, (arXiv:1609.09106), 9 2016.

[21] K. Kar, J. Kubilius, K. Schmidt, E. B. Issa, and J. J. DiCarlo. Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior. *Nature Neuroscience*, 22(6):974–983, 6 2019.

[22] T. Karaletsos, P. Dayan, and Z. Ghahramani. Probabilistic meta-representations of neural networks. *arXiv preprint*, (arXiv:1810.00555), 10 2018.

[23] K. Katahira. The relation between reinforcement learning parameters and the influence of reinforcement history on choice behavior. *Journal of Mathematical Psychology*, 66:59–69, 6 2015.

[24] M. Koivisto, H. Railo, A. Revonsuo, S. Vanni, and N. Salminen-Vaparanta. Recurrent processing in V1/V2 contributes to categorization of natural scenes. *The Journal of Neuroscience*, 31(7):2488–2492, 2 2011.

[25] N. Kriegeskorte. Deep neural networks: A new framework for modeling biological vision and brain information processing. *Annual Review of Vision Science*, 1(1):417–446, 11 2015.

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012.

[27] W.-H. Lin, A. M. Clarke, K. S. Pilz, M. H. Herzog, and M. Kunchulia. Intact reinforcement learning in healthy ageing. *bioRXiv*, 5 2023.

[28] K. J. Miller, M. Eckstein, M. M. Botvinick, and Z. Kurth-Nelson. Cognitive model discovery via disentangled RNNs. In *Advances in Neural Information Processing Systems*, pages 61377–61394, 2023.

[29] D. J. Navarro, T. L. Griffiths, M. Steyvers, and M. D. Lee. Modeling individual differences using Dirichlet processes. *Journal of Mathematical Psychology*, 50(2):101–122, 4 2006.

[30] J. P. O'Doherty, A. Hampton, and H. Kim. Model-based fMRI and its application to reward learning and decision making. *Annals of the New York Academy of Sciences*, 1104(1):35–53, 5 2007.

[31] M. L. Pedersen, M. J. Frank, and G. Biele. The drift diffusion model as the choice rule in reinforcement learning. *Psychonomic Bulletin and Review*, 24(4):1234–1251, 12 2017.

[32] J. C. Peterson, D. D. Bourgin, M. Agrawal, D. Reichman, and T. L. Griffiths. Using large-scale experiments and machine learning to discover theories of human decision-making. *Science*, 372(6547):1209–1214, 6 2021.

[33] S. T. Radev, U. K. Mertens, A. Voss, L. Ardizzone, and U. Kothe. BayesFlow: Learning complex stochastic models with invertible neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 33(4):1452–1466, 4 2022.

[34] F. Rafiei, M. Shekhar, and D. Rahnev. The neural network RTNet exhibits the signatures of human perceptual decision-making. *Nature Human Behaviour*, 8(9):1752–1770, 7 2024.

[35] R. Rajalingham, E. B. Issa, P. Bashivan, K. Kar, K. Schmidt, and J. J. DiCarlo. Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. *The Journal of Neuroscience*, 38(33):7255–7269, 8 2018.

[36] R. Ratcliff and G. McKoon. The diffusion decision model: Theory and data for two-choice decision tasks. *Neural Computation*, 20(4):873–922, 4 2008.

[37] M. Rmus, T.-F. Pan, L. Xia, and A. G. E. Collins. Artificial neural networks for model identification and parameter estimation in computational cognitive models. *PLOS Computational Biology*, 20(5):e1012119, 5 2024.

[38] D. E. Rumelhart and J. L. McClelland. Learning internal representations by error propagation. In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*, pages 318–362. MIT Press, 1987.

[39] R. Schaeffer, M. Khona, L. Meshulam, and I. Fiete. Reverse-engineering recurrent neural network solutions to a hierarchical inference task for mice. In *Advances in Neural Information Processing Systems*, 2020.

[40] S. Sevy, K. E. Burdick, H. Visweswaraiah, S. Abdelmessih, M. Lukin, E. Yechiam, and A. Bechara. Iowa gambling task in schizophrenia: A review and new data in patients with schizophrenia and co-occurring cannabis use disorders. *Schizophrenia Research*, 92(1-3):74–84, 5 2007.

[41] H. Siegelmann and E. Sontag. On the computational power of neural nets. *Journal of Computer and System Sciences*, 50(1):132–150, 2 1995.

[42] H. F. Song, G. R. Yang, and X.-J. Wang. Reward-based training of recurrent neural networks for cognitive and value-based tasks. *eLife*, 6, 1 2017.

[43] M. Song, Y. Niv, and M. Bo Cai. Using recurrent neural networks to understand human reward learning. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, number 43, pages 1388–1394, 2021.

[44] C. J. Spoerer, T. C. Kietzmann, J. Mehrer, I. Charest, and N. Kriegeskorte. Recurrent neural networks can explain flexible trading of speed and accuracy in biological vision. *PLOS Computational Biology*, 16(10):e1008215, 10 2020.

[45] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, 1998.

[46] H. Tang, M. Schrimpf, W. Lotter, C. Moerman, A. Paredes, J. Ortega Caro, W. Hardesty, D. Cox, and G. Kreiman. Recurrent computations for visual pattern completion. *Proceedings of the National Academy of Sciences*, 115(35):8835–8840, 8 2018.

[47] I. Tolstikhin, O. Bousquet, S. Gelly, and B. Schoelkopf. Wasserstein auto-encoders. *arXiv preprint*, (arXiv:1711.01558), 11 2017.

[48] D. Tuzsus, A. Brands, I. Pappas, and J. Peters. Exploration–exploitation mechanisms in recurrent neural networks and human learners in restless bandit problems. *Computational Brain & Behavior*, 5 2024.

[49] R. van den Bos, J. Homberg, and L. de Visser. A critical review of sex differences in decision-making tasks: Focus on the Iowa gambling task. *Behavioural Brain Research*, 238(1):95–108, 2 2013.

[50] D. Vickers. Evidence for an accumulator model of psychophysical discrimination. *Ergonomics*, 13(1):37–58, 4 2007.

[51] E.-J. Wagenmakers and S. Brown. On the linear relation between the mean and the standard deviation of a response time distribution. *Psychological Review*, 114(3):830–841, 7 2007.

[52] R. Wang, Y. Shen, P. Tino, A. Welchman, and Z. Kourtzi. Learning predictive statistics: strategies and brain mechanisms. *The Journal of Neuroscience*, 37(35):0144–17, 2017.

[53] R. C. Wilson and A. G. Collins. Ten simple rules for the computational modeling of behavioral data. *eLife*, 8:1–33, 2019.

[54] D. L. K. Yamins and J. J. DiCarlo. Using goal-driven deep learning models to understand sensory cortex. *Nature Neuroscience*, 19(3):356–365, 3 2016.

[55] G. R. Yang, M. R. Joglekar, H. F. Song, W. T. Newsome, and X.-J. Wang. Task representations in neural networks trained to perform many cognitive tasks. *Nature Neuroscience*, 22(2):297–306, 2 2019.

[56] E. Yechiam, J. R. Busemeyer, J. C. Stout, and A. Bechara. Using cognitive models to map relations between neuropsychological disorders and human decision-making deficits. *Psychological Science*, 16(12):973–978, 12 2005.

Acknowledgements

This work was supported in part by the Japan Society for the Promotion of Science (JSPS) KAKENHI, grant number 22H05163 and 24K15047, and Japan Science and Technology Agency (JST) Advanced International Collaborative Research Program (AdCORP), grant number JPMJKB2307.

Author contributions

H.H. designed and performed the research, collected and analyzed the data, and drafted and edited the paper.

Competing interests

The authors declare no competing interests.